ขจร โรจนเมธินทร์

Visual Basic 3.0

: ถึงคราวต้องสร้างปฏิทินเอง

เทคนิคการเขียนโปรแกรมสร้างปฏิทินมาใช้เอง โดยอาศัยโปรแกรม VBCalendar

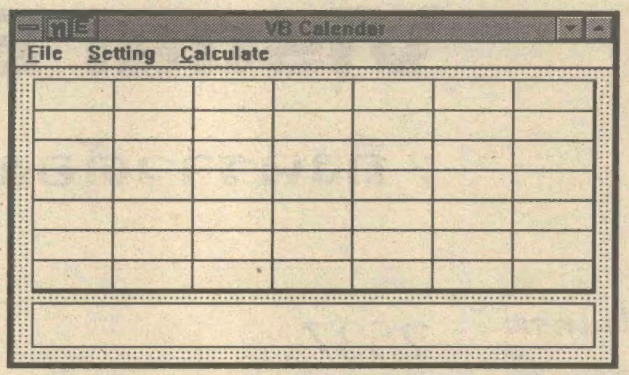
เมื่อ 2-3 วันก่อน ผู้เขียนมีความจำเป็นต้องค้นหาว่าวันที่บางวันในปี พ.ศ. 2519 ตรงกับวันใดของสัปดาห์ ตอนแรกคิดว่าจะใช้โปรแกรม Calendar ของวินโดวส์ แต่ก็นึกขึ้นมาได้ว่าในวินโดวส์ไทยเอดิชันนี้ไม่สามารถกำหนดปี พ.ศ. ที่น้อยกว่า 2523 (ค.ศ. 1980) ได้ เพราะติดปัญหามาจากดอส จะกลับไปใช้โปรแกรมไซด์คิกบนดอสก็ไม่รู้ว่าไปเก็บไว้ที่ใด ผู้เขียนก็เลยต้องเขียนโปรแกรมสร้างปฏิทินมาใช้เสียเอง ดังนั้นจึงถือโอกาสนำมาฝากผู้อ่าน เพื่อจะได้นำไปใช้ต่อ เพราะในโปรแกรม VB Calendar นี้ก็มีเทคนิคการเขียนโปรแกรมบางอย่างที่น่าสนใจไม่น้อย

โปรแกรม VB Calendar นี้จะคำนวณหาวันที่ของปฏิทินตามปีที่ผู้ใช้กำหนดเองโดยไม่ต้องพึ่งพาดอส ทำให้การคำนวณปฏิทินของปี ค.ศ. ทำได้ไม่จำกัด จะเรียกว่าเป็นปฏิทิน 100 ปีก็ได้ ในโปรแกรมนี้สามารถพิมพ์ปฏิทินออกมาทางเครื่องพิมพ์เป็นตารางเพื่อใช้ลงกำหนดการในแต่ละเดือนได้ด้วย

สร้างฟอร์ม VB Calendar ก่อน

ฟอร์ม VB Calendar ที่จะสร้างนี้จะต้องมีตารางที่ใช้แสดงเป็นปฏิทิน ในวิชวลเบสิกนั้นมีคอนโทรล Grid ซึ่งเราสามารถนำมาใช้เป็นตารางได้สะดวกมาก เพราะหากใช้คอนโทรล Grid จะสะดวกในแง่การจัด Alignment ของวันที่ในตารางเพียงแต่กำหนดพรอพเพอร์ตี้บางตัวเท่านั้น หากเราไม่ใช้ Grid นี้เราจะต้องมานั่งวาดตารางเองโดยใช้คอนโทรล Line นอกจากนี้ยังต้องใช้คอนโทรล Label อย่างน้อยก็ 25 อันมาใช้สำหรับแสดงวันที่ในตาราง และยังต้องจัดให้แต่ละเลเบลเรียงแถวตรงกัน ซึ่งค่อนข้างเสียเวลา การทำงานของโปรแกรมในช่วงใส่วันที่ก็จะช้ากว่าใช้ Grid จากรูปที่ 1 จะแสดงคอนโทรลที่ประกอบอยู่ในฟอร์ม VB Calendar ผู้เขียนใช้คอนโทรล Grid เป็นตารางขนาด 5 แถว 7 คอลัมน์สำหรับแสดงเป็นปฏิทิน กำหนดสีแบคกราวนด์ใหม่เป็นสีฟ้าและฟอร์กราวนด์เป็นสีแดง

เลือกใช้ฟอนต์แบบ TimesNew Roman ขนาด 12 คอนโทรลที่ใช้อีกอย่างก็คือ คอนโทรล Shape ซึ่งในที่นี้เป็นรูปสี่เหลี่ยมอยู่ใต้ตาราง และมีคอนโทรล Label ซ้อนอยู่ในคอนโทรล Shape ใช้แสดงวันที่ เดือน และปี ค.ศ. ปัจจุบัน โดยกำหนดฟอนต์และสีแบคกราวนด์เหมือน Grid แต่ฟอร์กราวนด์ใช้สีน้ำเงินแทน สำหรับค่าเริ่มต้นของคอนโทรลต่าง ๆ นี้แสดงอยู่ในตารางที่ 1



รูปที่ 1
แสดงคอนโทรล
ที่ประกอบอยู่ในฟอร์ม
VB Calendar

ออปเจ็กต์	พรอพเพอร์ตี้	ค่าเริ่มต้น
Form	Name	fCard
	AutoRedraw	True
	BorderStyle	1-Fixed Single
	Caption	"VB Calendar"
	MaxButton	False
Common dialog	Name	CMD
Labe	Name	dYear
	Alignment	1-Right justify
	Caption	""
Label	Name	dMonth
	Caption	""
Label	Name	cDay
	Alignment	2-Center
	Caption	""
Grid	Name	grdMnth
	Cols	7
	FixedCols	0
	FixedRows	0
	HighLight	False
	Rows	7
	ScrollBars	0-None
Shape	Name	Shape1
	FillStyle	0-Solid

ตารางที่ 1
แสดงคอนโทรลต่าง ๆ
ที่ประกอบเป็นฟอร์ม
VB Calendar

เมนูของโปรแกรม

จากรูปที่ 1 ซึ่งแสดงเมนูของ VB Calendar จะมีเมนูหลักในเมนูบาร์อยู่ 3 รายการคือ File, Setting และ Calculate ในเมนู File นี้จะจัดการในเรื่องการพิมพ์ปฏิทิน การเซตค่าเครื่องพิมพ์ เมนู Setting ใช้กำหนดเดือน ปี ค.ศ. ที่จะให้ปฏิทินแสดง และเมนู Calculate ใช้คำนวณระหว่างค่า ปี ค.ศ. เป็น ปี พ.ศ.หรือกลับกัน ค่าเริ่มต้นของเมนูแสดงอยู่ในตารางที่ 2

หัวใจของ VB Calendar

ความจริงแล้วอาจจะเรียกโค้ดส่วนที่จะกล่าวถึงนี้ว่าเป็นโค้ดส่วนสำคัญที่สุดของโปรแกรมก็ว่าได้ เพราะการทำงานทั้งหมดจะต้องอาศัยโค้ดนี้ทั้งนั้น โค้ดส่วนนี้เราจะเขียนไว้ในส่วน declaration ของฟอร์ม CARD.FRM (รูปที่ 2)

ในโค้ดรูปที่ 2 นี้ประกอบด้วยโพรซีเดอร์ ClearGrid, DayOfMonth, Find1stDay และ FillGrid ผู้เขียนจะขอกล่าวถึงโพรซีเดอร์ FillGrid ก่อน โพรซีเดอร์นี้มีหน้าที่หลักก็คือ เขียนวันที่ลงในกริดตามลำดับที่ถูกต้องของเดือนในปีที่ต้องการเริ่มต้นโดยการเซตค่าตัวแปร

Caption	Name	Shortcut Key
"&File"	fileMain	
– "&Print"	mPrint	
– "Print &Setup"	mSetup	
– "-"	mfSep	
– "&About..."	mAbout	
– "-"	mfSep1	
– "E&xit"	mExit	
"&Setting"	optionMain	
– "&Month"	mMonth	F4
– "&Year"	mYear	F5
– "&Previous Month"	mPrev	Shift+F7
– "&Next Month"	mNext	Shift+F8
– "&Current"	mCur	Shift+F9
"&Calculate"	calMain	
– "To &Thai"	mCal	F3

ตารางที่ 2
แสดงค่าเริ่มต้น
ของเมนูโปรแกรม
VB Calendar

```
Sub ClearGrid ()
    Dim r, rs As Integer

    r = 1
    rs = grdMnth.Rows
    grdMnth.Rows = rs + 1
    While r < rs
      grdMnth.RemoveItem 1
      r = r + 1
    Wend
    grdMnth.Rows = 1
End Sub

Function DayOfMonth (mm, yy As Integer) As Integer
    Select Case mm
      Case 1, 3, 5, 7, 8, 10, 12
        DayOfMonth = 31
      Case 4, 6, 9, 11
        DayOfMonth = 30
      Case 2
        DayOfMonth = 28
        If (yy Mod 4) = 0 Then DayOfMonth = 29
    End Select
End Function

Sub FillGrid ()
    Dim loc1st As Integer
    Dim rowTxt As String
    Dim colTxt As String
    Dim i, d, n As Integer

    i = 1
    d = 1
    rowTxt = ""
    loc1st = Find1stDay(gMonth, gYear)
    n = DayOfMonth(gMonth, gYear)
    While d <= n
      If i < loc1st Then
        colTxt = ""
      Else
        colTxt = Format(d, "@@")
        d = d + 1
      End If
      rowTxt = rowTxt & colTxt & Chr$(9)
      If (i Mod 7) = 0 Then
        grdMnth.AddItem rowTxt
        rowTxt = ""
      ElseIf d = n + 1 Then
        grdMnth.AddItem rowTxt
      End If
      i = i + 1
    Wend
    SetGridAlign
    grdMnth.FixedRows = 1
    grdMnth.Col = 0
    grdMnth.Row = 1
End Sub


Function Find1stDay (cmm, cyy As Integer) As Integer
    Dim ii, ij As Integer

    ii = (cyy + (cyy - 1) \ 4) Mod 7
    Select Case cmm
      Case 2, 3, 11
        ii = (ii + 3) Mod 7
      Case 4, 7
        ii = (ii + 6) Mod 7
      Case 5
        ii = (ii + 1) Mod 7
      Case 6
        ii = (ii + 4) Mod 7
      Case 8
        ii = (ii + 2) Mod 7
      Case 9, 12
        ii = (ii + 5) Mod 7
      Case 10
        ii = (ii + 7) Mod 7
    End Select
    ij = cyy Mod 4
    If (ij = 0) And (cmm > 2) Then
      ii = (ii + 1) Mod 7
    End If
    If (ii = 0) Then ii = 7
    Find1stDay = ii
End Function
```

รูปที่ 2

ต่าง ๆ ที่ต้องใช้ในโพรซีเดอร์ ได้แก่ i จะใช้นับตำแหน่งเซลล์ในกริด, d ใช้นับวันที่ของเดือน, loc1st ใช้เก็บลำดับวันเริ่มแรกของเดือน, n เก็บจำนวนวันของเดือน, rowTxt ใช้เป็นตัวแปรที่เก็บสตริงที่จะใส่ในกริด สำหรับ Find1stDay นั้นใช้คำนวณว่าวันที่ 1 ของเดือนที่ต้องการนั้นตรงกับวันใด เช่น ถ้าจะแสดงปฏิทินของเดือนสิงหาคม ค.ศ. 2000 โพรซีเดอร์ Find1stDay ก็จะต้องหาว่าวันที่ 1 ของเดือนสิงหาคมนี้ตรงกับวันอังคาร โดยจะคืนค่ามาเป็น 3 (อาทิตย์เป็น 1,...,เสาร์เป็น 7) เป็นต้น ส่วน DayOfMonth ก็มีหน้าที่เพียงตรวจว่าเดือนที่ต้องการมีกี่วัน โดยใช้คำสั่ง Select Case เท่านั้น แต่ต้องไม่ลืมคำนวณปีที่เดือนกุมภาพันธ์มี 29 วันด้วย

เมื่อกำหนดตัวแปรที่ต้องการได้หมดแล้ว ก็จะเริ่มต้นใส่วันที่ลงเซลล์ในกริด โดยเราจะใช้ลูป While...Wend วนลูปตามจำนวนวันของเดือนใน n สำหรับการใส่ค่าลงในเซลล์ของกริดนั้นจะทำทีละแถว โดยใช้ Method AddItem ในแถวที่เราใส่นี้ค่าของแต่ละเซลล์จะแยกกันด้วย Chr$(9) หรือ Tab นั่นเอง (สำหรับเรื่องการใช้คอนโทรลกริดนี้ยังมีเทคนิคอีกหลายอย่าง ผู้เขียนจะนำมากล่าวในตอนต่อ ๆ ไป) ดังนั้นเราจะต้องสร้างสตริง rowTxt สำหรับแต่ละแถวให้ถูกต้อง คือถ้ายังไม่ใช้ตำแหน่งวันที่ 1 ของเดือนนั้น ค่าของเซลล์จะต้องเป็น "" แต่ถ้าเป็นตำแหน่งที่ถูกต้องแล้วก็จะสร้างค่าเซลล์เป็นสตริงของวันที่จาก Format (d," @@") ในการเติมค่าเซลล์ทั้งแถวนั้นจะทำเมื่อสร้าง rowTxt ครบ 7 วันแล้ว คือ (i Mod 7) ต้องเท่ากับ 0 แต่ถ้าเป็นกรณีสัปดาห์สุดท้ายของเดือนก็จะต้อง Add Item เข้าไปทันที หลังจากใส่ค่าในเซลล์ของกริดเรียบร้อยแล้ว ก็เรียก SetGrid Align (ดูโค้ดในรูปที่ 3) เพื่อจะกำหนด Alignment ของคอลัมน์กริดให้เป็นกลางเซลล์ และกำหนดจำนวนแถวที่จะอยู่คงที่ในกริด โดยกำหนดค่าFixedRows และกำหนดตำแหน่งเซลล์ปัจจุบันผ่านค่า Col และ Row ของgrdMnth

โมดูลสำหรับกำหนดค่าที่จำเป็น

จะเห็นว่าใน FillGrid อ้างถึงตัวแปร gMonth และ gYear ซึ่งเป็นตัวแปรที่เราประกาศอยู่ในโมดูล VBCARD.BAS (ดูโค้ดในรูปที่ 4) ตัวแปรทั้งสองนี้จะเก็บค่าเดือนและปี ค.ศ. ที่ต้องการให้แสดงปฏิทิน นอกจากนี้ยังมีตัวแปร cMonth, cYear ที่เก็บค่าเดือน ปี ค.ศ. ปัจจุบันไว้

```
Sub SetGridAlign ()
   Dim i As Integer

   For i = 0 To grdMnth.Cols - 1
     grdMnth.ColAlignment(i) = 2       'Center Text
     grdMnth.FixedAlignment(i) = 2
   Next i
End Sub
```

รูปที่ 3

```
Option Explicit

Global cMonth, cYear As Integer
Global gMonth, gYear As Integer
```

รูปที่ 4

```
Global stMonth(12) As String

Global Const DLG_PRINT = 5
Global Const PD_PRINTSETUP = &H40
Global Const LEFT_BUTTON = 1
Global Const RIGHT_BUTTON = 2

Global Const MODAL = 1
Global Const WF_CPU286 = &H2
Global Const WF_CPU386 = &H4
Global Const WF_CPU486 = &H8
Global Const WF_STANDARD = &H10
Global Const WF_ENHANCED = &H20
Global Const WF_80x87 = &H400

Declare Function GetWindowsDirectory% Lib "Kernel" (ByVal lpBuffer
As String, ByVal nSize%)
Declare Function GetWinFlags& Lib "Kernel" ()
Declare Function GetFreeSpace& Lib "Kernel" (ByVal wFlags%)

Sub InitStMonth ()
   stMonth(1) = "January"
   stMonth(2) = "February"
   stMonth(3) = "March"
   stMonth(4) = "April"
   stMonth(5) = "May"
   stMonth(6) = "June"
   stMonth(7) = "July"
   stMonth(8) = "August"
   stMonth(9) = "September"
   stMonth(10) = "October"
   stMonth(11) = "November"
   stMonth(12) = "December"
End Sub
```

รูปที่ 5

นอกจากนี้ก็ยังมีตัวแปรและค่าคงที่บางอย่างที่จำเป็นต้องใช้กับคอนโทรล Common dialog ได้แก่ DLG_PRINT ซึ่งใช้ระบุชนิดไดอะล็อกบ็อกซ์ ในที่นี้เลือกไดอะล็อกบ็อกซ์เกี่ยวกับ Printer, PD_PRINTSETUP เป็นแฟล็กที่ใช้กำหนดว่าจะใช้ไดอะล็อก PrintSetup ของวินโดวส์ ตัวแปรสตริงอะเรย์ stMonth เราจะใช้เก็บชื่อเดือนแบบเต็มทั้ง 12 ชื่อ และใช้สับรูทีน InitStMonth เป็นตัวเซตค่าสำหรับตัวแปรและการประกาศฟังก์ชันอื่นในโค้ดรูปที่ 5 จะใช้สำหรับฟอร์ม fAbout เหมือนในตอนก่อน ๆ ซึ่งจะไม่ขอกล่าวถึง

เหตุการณ์ขณะโหลดฟอร์ม VB Calendar

เมื่อโปรแกรม VB Calendar ทำงานขณะที่โหลดฟอร์มเราจะต้องจัดการทำงานหลายอย่างเพื่อให้แสดงปฏิทินในกริดได้ดังรูปที่ 6

```
Sub Form_Load ()
    cMonth = Month(Date)
    cYear = Year(Date)
    gMonth = cMonth
    gYear = cYear
    InitStMonth
    dMonth.Caption = stMonth(gMonth)
    dYear.Caption = gYear
    cDay.Caption = Day(Date)
    FillHeader
    FillGrid
    grdMnth.FixedRows = 1
    grdMnth.Col = 0
    grdMnth.Row = 1
    Left = (Screen.Width - Width) \ 2
    Top = (Screen.Height - Height) \ 2
End Sub
```

รูปที่ 6

```
Sub FillHeader ()
    Dim i As Integer
    ReDim stDay(7) As String

    stDay(1) = "Sun"
    stDay(2) = "Mon"
    stDay(3) = "Tue"
    stDay(4) = "Wed"
    stDay(5) = "Thu"
    stDay(6) = "Fri"
    stDay(7) = "Sat"
    rowTxt = ""
    For i = 1 To 7
      rowTxt = rowTxt & stDay(i) & Chr$(9)
    Next i
    grdMnth.FixedRows = 0
    grdMnth.FixedCols = 0
    grdMnth.Rows = 1
    grdMnth.Cols = 1
    grdMnth.AddItem rowTxt
    grdMnth.RemoveItem 0
End Sub
```

รูปที่ 7

VB Calendar

File Setting Calculate

Sun	Mon	Tue	Wed	Thu	Fri	Sat
					1	2
3	4	5	6	7	8	9
10	11	12	13	14	15	16
17	18	19	20	21	22	23
24	25	26	27	28	29	30
31						

July 24 1994

รูปที่ 8 แสดงจอภาพเมื่อโหลดฟอร์ม VB Calendar แล้ว

ในโค้ดรูปที่ 6 เราจะเริ่มเก็บค่าเดือนปีปัจจุบันไว้ในตัวแปร cMonth, cYear ก่อน เขียนวันที่ เดือน ค.ศ. ปัจจุบันไว้ที่ Caption ของเลเบล cDay, dMonth, dYear เรียก FillHeader (ดูโค้ดรูปที่ 7) เพื่อใส่ชื่อวัน Sun,...,Sat ไว้ที่แถวแรกของกริด จากนั้นจึงเขียนวันที่ในเดือนนั้นลงในกริดด้วย FillGrid กำหนดแถวคงที่โดยกำหนดค่า FixedRows และจัดให้ฟอร์มโหลดอยู่กลางจอภาพ (ดูรูปที่ 8)

จากโค้ดในรูปที่ 7 ก็จะคล้ายกับการใส่ค่าเซลล์ใน FillGrid เพียงแต่ใส่เป็นสตริงชื่อวันเท่านั้น หลังจากนั้นจะกำหนดให้ไม่มีแถวที่อยู่คงที่ในกริด กำหนดจำนวนแถวและคอลัมน์ในกริดใหม่ให้เป็น 1 เท่านั้นเพื่อที่จะได้ AddItem ที่แถวสุดท้ายของกริดเท่านั้น และ FillGrid ก็จะ

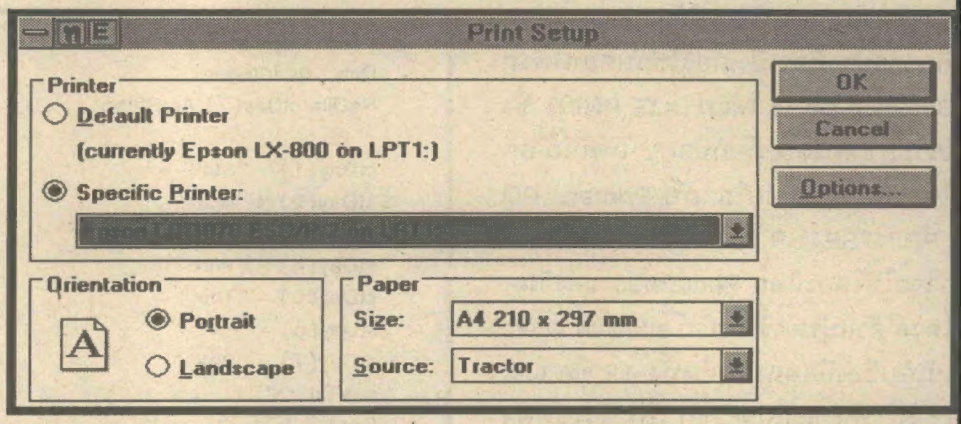
เพิ่มแถวของกริดที่แถวสุดท้ายโดยไม่มีแถววางเกินมา

เหตุการณ์ที่เกิดจากเมนู

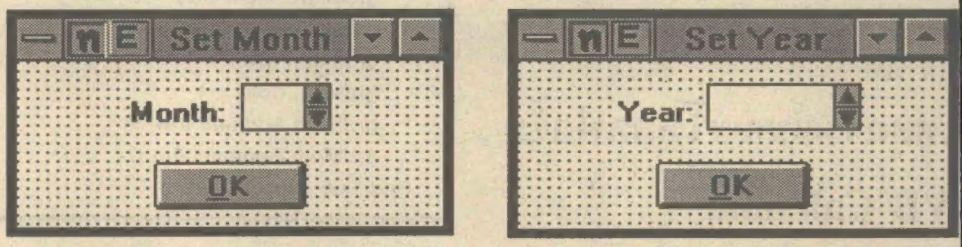
เมนูแรกที่จะกล่าวถึงคือเมนู File ซึ่งประกอบด้วยคำสั่ง Print, Print Setup, About และ Exit แต่ตอนนี้จะกล่าวถึงเฉพาะ 3 คำสั่งหลังก่อน

```
Sub mSetup_Click ()
    On Error Resume Next
    CMD.Flags = PD_PRINTSETUP
    CMD.Action = DLG_PRINT
End Sub

Sub mAbout_Click()
    fAbout.Show MODAL
End Sub

Sub mExit_Click()
    End
End Sub
```

รูปที่ 9



รูปที่ 10 แสดงไดอะล็อกบ็อกซ์ร่วม Print Setup



รูปที่ 11 แสดงส่วนประกอบในฟอร์ม fMonth และ fYear

จากโค้ดรูปที่ 9 เหตุการณ์ mSetup_Click นั้นจะเรียกใช้ไดอะล็อกบ็อกซ์กลางของวินโดวส์โดยเลือกใช้ไดอะล็อก Print Setup (ดูรูปที่10) เรามีหน้าที่เพียงกำหนดค่า Flags และ Action เท่านั้น ผู้อ่านก็สามารถจะกำหนดค่าต่าง ๆ ที่จะใช้ในการพิมพ์ได้

เมนูต่อไปที่จะพูดถึงคือเมนู Setting มีคำสั่ง Month, Year, Previous Month, Next Month และ Current หน้าที่ของเมนูนี้ก็คือ เซตค่าตัวแปร gMonth, gYear และ Caption ของ dMonth, dYear แล้วเรียก ClearGrid, FillGrid เพื่อเติมวันที่ในกริด โดยใน mCur_Click นั้นเราจะนำค่าเดือนและปีปัจจุบันมาใช้เพื่อจะแสดงปฏิทิน mNext_Click จะเพิ่มค่า gMonth ขึ้นหนึ่งค่า ถ้าค่าไม่เกิน 12 ก็จะแสดงปฏิทิน, mPrev_Click จะลดค่า gMonth ลงหนึ่ง ถ้าไม่ต่ำกว่า 1 สำหรับเหตุการณ์ mMonth_Click และmYear_Click เราต้องโหลดฟอร์ม fMonth, fYear (ดูรูปที่ 11) เพื่อจะรับค่าเดือน หรือปี ค.ศ. ที่ต้องการก่อน (ดูโค้ดรูปที่ 12) ซึ่งผู้เขียนจะกล่าวถึงต่อไป

เมนูสุดท้ายของโปรแกรม VB Calendar คือ Calculate ซึ่งมีคำสั่งเดียวคือ To Thai ซึ่งมีหน้าที่คำนวณปีจาก ค.ศ. เป็น พ.ศ. หรือกลับกันก็ได้ โดยหน้าที่ของเมนูเพียงแต่โหลดฟอร์ม fCalc มาทำงานเท่านั้น

ฟอร์ม fMonth และ fYear

ถ้าหากผู้อ่านกดคีย์ F4 หรือ F5 ก็จะมีฟอร์ม fMonth และ fYear แสดงตามลำดับ (ดูรูปที่ 11) สำหรับส่วนประกอบของฟอร์มและค่าเริ่มต้นแสดงที่ต้องสร้างแสดงอยู่ในตารางที่ 3 และ 4

จะเห็นว่าเราใช้คอนโทรล Spin Button ซึ่งมีให้ในวิชวลเบสิก 3.0 เฉพาะ Professional Edition เท่านั้น ซึ่งในโปรแกรมนี้เราจะใช้ Spin button เพื่อควบคุมการเปลี่ยนแปลงค่าของตัวแปร gMonth และ gYear ทีละหนึ่ง โดยถ้าผู้อ่านคลิ๊กที่บัททอนลูกศรบนก็จะเป็นการเพิ่มค่าทีละหนึ่ง แต่ถ้าคลิ๊กเม้าส์ที่ลูกศรลงก็จะลดค่าทีละหนึ่ง โพรซีเดอร์ที่จัดการเหตุการณ์เมื่อเราคลิ๊กที่บัททอนคือ Spin1_SpinUp และ Spin1_ SpinDown จากโค้ดรูปที่ 13 ได้แสดงเฉพาะ Spin1 ของฟอร์ม fMonth เท่านั้น ถ้าเป็นของฟอร์ม fYear นั้นเราก็ไม่จำเป็นต้องตรวจสอบค่า gYear โดยเราสามารถเปลี่ยนแปลงค่าได้ทันที

```
Sub mCur_Click ()
    gMonth = cMonth
    gYear = cYear
    dMonth.Caption = stMonth(gMonth)
    dYear.Caption = gYear
    ClearGrid
    FillGrid
End Sub

Sub mMonth_Click ()
    fMonth.Show MODAL
    dMonth.Caption = stMonth(gMonth)
    dYear.Caption = gYear
    ClearGrid
    FillGrid
End Sub

Sub mNext_Click ()
    If gMonth < 12 Then
      gMonth = gMonth + 1
      dMonth.Caption = stMonth(gMonth)
      dYear.Caption = gYear
      ClearGrid
      FillGrid
    End If
End Sub

Sub mPrev_Click ()
    If gMonth > 1 Then
      gMonth = gMonth - 1
      dMonth.Caption = stMonth(gMonth)
      dYear.Caption = gYear
      ClearGrid
      FillGrid
    End If
End Sub

Sub mYear_Click ()
    fYear.Show MODAL
    dMonth.Caption = stMonth(gMonth)
    dYear.Caption = gYear
    ClearGrid
    FillGrid
End Sub

Sub mCal_Click ()
    fCalc.Show MODAL
End Sub
```

รูปที่ 12

ออปเจ็กต์	พรอพเพอร์ตี้	ค่าเริ่มต้น
Form	Name	fMonth
	BorderStyle	3-Fixed double
	Caption	"Set Month"
	MaxButton	False
	MinButton	False
Command Button	Name	cmdOK
	Caption	"&OK"
Label	Name	lblMM
Spin Button	Name	Spin1
Text Box	Name	txtMM
	Text	""

ตารางที่ 3
แสดงส่วนประกอบ
และค่าเริ่มต้น
ของฟอร์ม fMonth

ออปเจ็กต์	พรอพเพอร์ตี้	ค่าเริ่มต้น
Form	Name	fYear
	BorderStyle	3-Fixed double
	Caption	"Set Year"
	MaxButton	False
	MinButton	False
Command Button	Name	cmdOK
	Caption	"&OK"
Label	Name	lblYY
Spin Button	Name	Spin1
Text Box	Name	txtYY
	Text	""

ตารางที่ 4
แสดงส่วนประกอบ
และค่าเริ่มต้น
ของฟอร์ม fYear

นอกจากนี้เรายังต้องเขียนโค้ดส่วนประกอบอื่น ๆ เพื่อให้ฟอร์มทั้งสองทำงานได้ดีขึ้นได้ แสดงไว้ในโค้ดรูปที่ 14 (แสดงเฉพาะฟอร์ม fMonth) คือ Form_Load จะให้ฟอร์มอยู่กลางจอและนำค่า gMonth และ gYear มาแสดงที่เท็กซ์-บอกซ์, txtMM_Change จะนำค่า Text ของเท็กซ์บ็อกซ์ txtMM เมื่อมีการเปลี่ยนแปลงมากำหนดให้ gMonth, txtMM_GotFocus จะเป็นกรณีที่เราเลื่อนแท็ปมาอยู่ที่คอนโทรลเท็กซ์บ็อกซ์ก็จะกำหนดเท็กซ์เป็น Highlight โดยกำหนดที่ SelStart ให้เป็นจุดเริ่มต้นของเท็กซ์คือ 0 และ SelLength ให้ยาวเท่ากับความยาวเท็กซ์ทั้งหมด, cmdOK_Click และ txtMM_KeyPress ทำหน้าที่คล้ายกันคืออันโหลดฟอร์ม fMonth และ fYear โดยใช้ตัวแปร Me อ้างถึงฟอร์มปัจจุบันขณะนั้น

ฟอร์ม fCalc

ฟอร์ม fCalc นี้จะรับผิดชอบต่อการแปลงจากปี พ.ศ. เป็น ค.ศ. ผู้อ่านจะต้องสร้างฟอร์มขึ้นก่อนตามตารางที่ 5 และเมื่อเราเรียกฟอร์มจากเมนู Calculate ก็จะปรากฏฟอร์มตามรูปที่ 15

โค้ดที่จะรับผิดชอบการทำงานของฟอร์ม fCalc แสดงอยู่ในโค้ดรูปที่ 16

```
Sub Spin1_SpinDown ()
    If gMonth > 1 Then
      gMonth = gMonth - 1
      txtMM.Text = Format(gMonth, "##")
    End If
End Sub

Sub Spin1_SpinUp ()
    If gMonth < 12 Then
      gMonth = gMonth + 1
      txtMM.Text = Format(gMonth, "##")
    End If
End Sub
```

รูปที่ 13

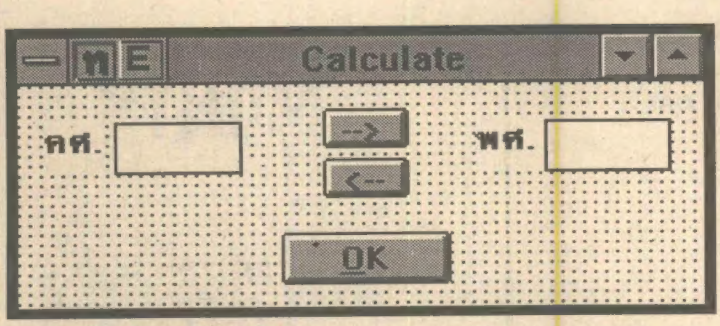
```
Sub cmdOK_Click ()
    Unload Me
End Sub

Sub Form_Load ()
    txtMM.Text = Format(gMonth, "##")
    Left = (Screen.Width - Width) \ 2
    Top = (Screen.Height - Height) \ 2
End Sub

Sub txtMM_Change ()
    gMonth = Val(txtMM.Text)
    If gMonth = 0 Then
      gMonth = cMonth
    End If
End Sub

Sub txtMM_GotFocus ()
    txtMM.SelStart = 0
    txtMM.SelLength = Len(txtMM.Text)
End Sub

Sub txtMM_KeyPress (KeyAscii As Integer)
    If KeyAscii = 13 Then
      KeyAscii = 0
      Unload Me
    End If
End Sub
```

รูปที่ 14

นั้นผู้เขียนจะขอกล่าวถึงเฉพาะบัททอน "<–" และ "–>" ที่ทำหน้าที่คำนวณจาก พ.ศ. เป็น ค.ศ. โดยลบด้วย 543 แต่ถ้าจาก ค.ศ. เป็น พ.ศ. ก็จะต้องบวกด้วย 543 นอกจากนั้นก็ต้องจัดการเหตุการณ์ Change ของเท็กซ์บ็อกซ์ทั้งสองที่เราใช้กำหนดให้ค่า Text ของเท็กซ์บ็อกซ์ตรงข้ามเป็น "" ถ้าค่า Text ของตัวเองเป็น "" และใช้ GotFocus เพื่อเซต Highlight ของเท็กซ์

โพรซีเดอร์สำหรับเลเบล dMonth และ dYear

ปกติในโปรแกรมที่ผ่านมาเราไม่ค่อยเขียนโพรซีเดอร์สำหรับเลเบลเลยแต่ในบทความนี้เราจะเขียนโพรซีเดอร์รับผิดชอบถ้าผู้ใช้โปรแกรมคลิ๊กเม้าส์บนเลเบล dMonth หรือ dYear หากคลิ๊กเม้าส์ปุ่มซ้ายก็จะเป็นการลดค่าลงหนึ่ง แต่ถ้าคลิ๊กปุ่มขวาจะเพิ่มค่าขึ้นหนึ่ง ซึ่งสำหรับเลเบล dMonth นั้นเราจะเรียกโพรซีเดอร์ mNext_click และ mPrev_Click มาใช้งานได้เลย (ดูโค้ดรูปที่ 17)



รูปที่ 15
แสดงส่วนประกอบในฟอร์ม fCalc

การตรวจสอบว่าผู้ใช้คลิ๊กเม้าส์ปุ่มใดเราจะต้องใช้เหตุการณ์ MouseDown ตรวจสอบโดยดูที่ค่า Button ซึ่งจะเป็น LEFT_BUTTON หรือ RIGHT_BUTTON แล้วแต่กรณี

การพิมพ์ตารางปฏิทิน

การใช้เครื่องพิมพ์ในวิชวลเบสิกนี้สะดวกมาก เนื่องจากวิชวลเบสิกกำหนดให้เครื่องพิมพ์เป็นออปเจ็กต์ชนิดหนึ่งที่เรียกใช้ได้อยู่แล้ว ในบทความตอนนี้จะกล่าวถึงเฉพาะคำสั่งที่จำเป็นก่อน

ตารางที่ 5
แสดงส่วนประกอบของฟอร์ม fCalc

ออปเจ็กต์	พรอพเพอร์ตี้	ค่าเริ่มต้น
Form	Name	fCalc
	BorderStyle	3-Fixed Double
	Caption	"Calculate"
	MaxButton	False
	MinButton	False
Command Button	Name	cmdCalc2
	Caption	"<—"
Command Button	Name	cmdCalc1
	Caption	"—>"
Command Button	Name	cmdOK
	Caption	"&OK"
Text Box	Name	txtTh
	Text	""
Text Box	Name	txtUs
	Text	""
Label	Name	lblTh
	Caption	"พศ."
Label	Name	lblUs
	Caption	"คศ."

```
Sub cmdCalc1_Click ()
    If txtUs.Text <> "" Then
      txtTh.Text = Format(Val(txtUs.Text) + 543, "####")
    End If
End Sub

Sub cmdCalc2_Click ()
    If txtTh.Text <> "" Then
      txtUs.Text = Format(Val(txtTh.Text) - 543, "####")
    End If
End Sub

Sub cmdOK_Click ()
    Unload Me
End Sub

Sub Form_Load ()
    Left = (Screen.Width - Width) \ 2
    Top = (Screen.Height - Height) \ 2
End Sub

Sub txtTh_Change ()
    If txtTh.Text = "" Then
      txtUs.Text = ""
    End If
End Sub

Sub txtTh_GotFocus ()
    txtTh.SelStart = 0
    txtTh.SelLength = Len(txtTh.Text)
End Sub

Sub txtUs_Change ()
    If txtUs.Text = "" Then
      txtTh.Text = ""
    End If
End Sub

Sub txtUs_GotFocus ()
    txtUs.SelStart = 0
    txtUs.SelLength = Len(txtUs.Text)
End Sub
```

รูปที่ 16

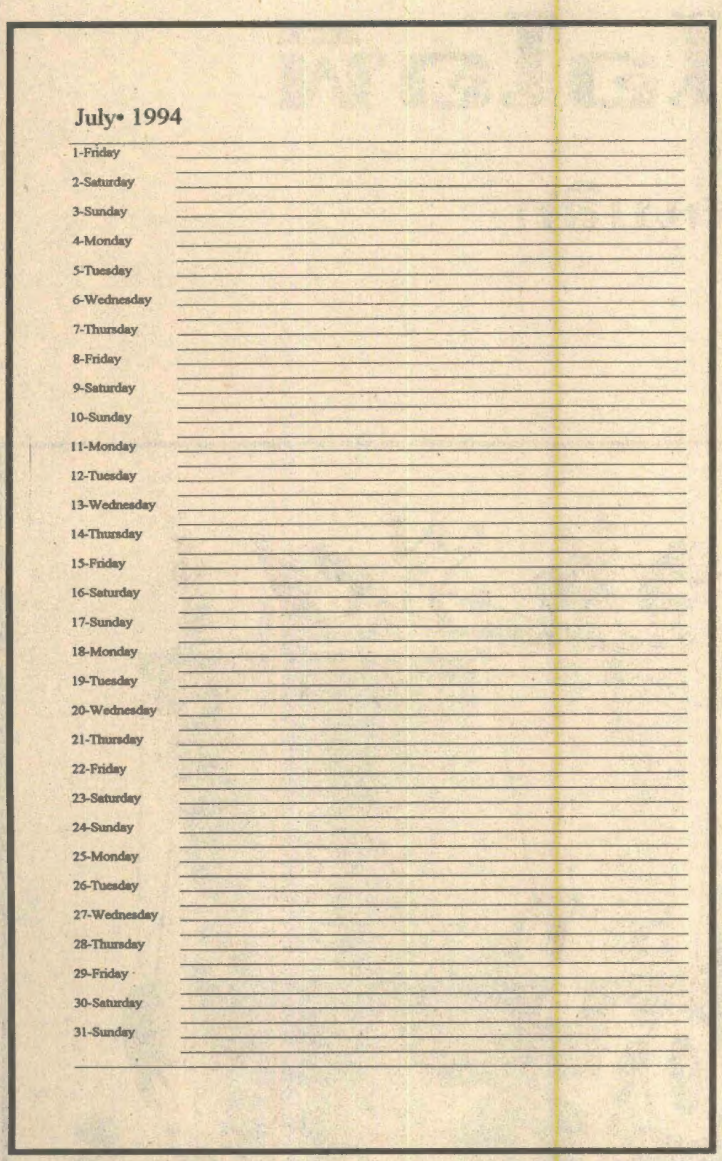
```
Sub dMonth_MouseDown (Button As Integer, Shift As Integer, X As
Single, Y As Single)
    Select Case Button
      Case LEFT_BUTTON
        mPrev_Click
      Case RIGHT_BUTTON
        mNext_Click
    End Select
End Sub

Sub dYear_MouseDown (Button As Integer, Shift As Integer, X As
Single, Y As Single)
    Select Case Button
      Case LEFT_BUTTON
        gYear = gYear - 1
      Case RIGHT_BUTTON
        gYear = gYear + 1
    End Select
    dMonth.Caption = stMonth(gMonth)
    dYear.Caption = gYear
    ClearGrid
    FillGrid
End Sub
```

รูปที่ 17

จากโค้ดรูปที่ 18 เราต้องกำหนดไดนามิกอะเรย์ขึ้นมาใช้งาน คือ Day Name(7) ซึ่งเราจะใช้เก็บรายชื่อวันแบบเต็มเพื่อใช้พิมพ์ การเรียกใช้เครื่องพิมพ์ก็เพียงแต่ใช้ออปเจ็กต์ Printer แล้วตามด้วยพรอพเพอร์ตี้ที่ต้องการ เริ่มด้วยเราจะต้องกำหนดชื่อฟอนต์ที่จะใช้พิมพ์ในที่

```
Sub mPrint_Click ()
    Dim ii, ij As Integer
    ReDim DayName(7) As String

    DayName(0) = "Sunday"
    DayName(1) = "Monday"
    DayName(2) = "Tuesday"
    DayName(3) = "Wednesday"
    DayName(4) = "Thursday"
    DayName(5) = "Friday"
    DayName(6) = "Saturday"
    Printer.FontName = "Times New Roman"
    Printer.FontSize = 18
    Printer.Print " "; stMonth(gMonth); Chr(149); gYear
    Printer.FontSize = 10
    Printer.FontUnderline = True
    Printer.Print Space(182)
    Printer.FontUnderline = False
    ij = Find1stDay(gMonth, gYear)
    For ii = 1 To DayOfMonth(gMonth, gYear)
      Printer.Print Format(ii, "@@"); "-"; DayName(ij);
      Printer.FontUnderline = True
      Printer.Print Tab(20); Space(150)
      Printer.Print Tab(20); Space(150)
      Printer.FontUnderline = False
      ij = ij + 1
      If ij > 6 Then ij = 0
    Next
    Printer.FontUnderline = True
    Printer.Print Space(182)
    Printer.FontUnderline = False
    Printer.EndDoc
End Sub
```

รูปที่ 18

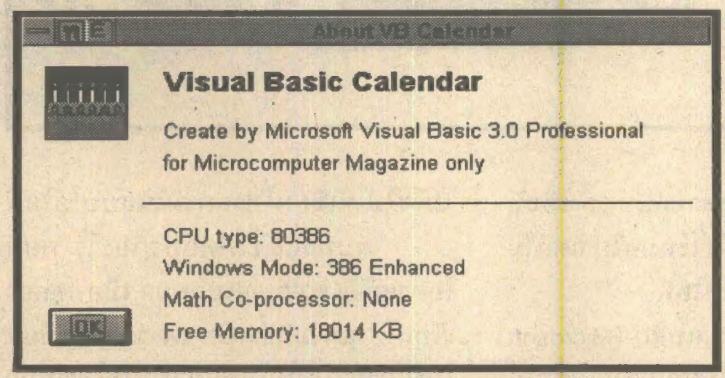


รูปที่ 19
แสดงตารางปฏิทิน
ที่พิมพ์จาก
VB Calendar

นี้ใช้ Times New Roman จากนั้นกำหนดขนาดที่ค่า FontSize ให้มีขนาด 18 แล้วจึงเริ่มพิมพ์ชื่อเดือนกับปีด้วย Method Print หลังจากนั้นเราจะกำหนดให้พิมพ์แบบตีเส้นเพื่อจะพิมพ์ช่องว่างโดยเราจะตีเส้นวันละสองบรรทัด (ดูรูปที่19) เมื่อเราพิมพ์เสร็จจะต้องใช้คำสั่ง Printer.EndDoc เพื่อสิ้นสุดการพิมพ์และส่งข้อมูลทั้งหมดออกทางเครื่องพิมพ์

ท้ายเรื่องนี้

บทความตอนนี้ผู้เขียนได้นำเอาคอนโทรลที่มีให้ใช้ในวิชวลเบสิก 3.0 มาเสนอ หวังว่าผู้อ่านคงจะได้แนวคิดในการนำกริดไปใช้งานได้ สำหรับการใช้ออป-เจ็กต์เครื่องพิมพ์นั้น คงจะมีโอกาสได้นำมาเสนอในรายละเอียดในตอนต่อไปอีกครั้ง



รูปที่ 20
แสดงจอภาพ
เมื่อเรียกคำสั่ง
About จากเมนู File